IN THE CLAIMS:

1. (currently amended) A method performed by a data processing system having a memory, comprising the steps of:

inputting a CCFG;

scheduling the CCFG to produce a scheduled CCFG;

selecting, while a first thread is running, a first node of the scheduled CCFG;

producing a first copy of the first node for an SCFG;

determining that the first node is in a second thread, different from the first thread;

coupling, to each of a plurality of nodes of the SCFG indicated by the first thread, a node that saves a representation, of a flow of control state of the first thread, in a first state variable, wherein the first state variable is not used to save a representation of a flow of control state of the second thread; and

coupling, as a successor node to each node that saves a representation, the same successor node for resuming a flow of control state of the second thread by reading a second state variable and choosing an edge to a node to be executed next in the second thread, wherein the successor node has an outgoing edge to each node that can be executed next in the second thread and the second state variable is not used to save a representation of a flow of control state of the first thread.

~~and~~

~~coupling, if a first thread of the first node is suspended, between a second node of the SCFG of a second previously-running thread and the first copy, a first context switch, wherein the context switch saves a second state, of the second previously-running thread, into a state variable dedicated to the second previously-running thread.~~

2. (currently amended) The method of claim 1, wherein each node that saves a representation, and the successor node, are ~~the first context switch is~~ comprised of code that saves a state of a thread being suspended in a state variable and that

resumes another thread by performing a multiway branch on a state variable for a thread being resumed.

3. (original) The method of claim 1, wherein ~~the translation of the CCFG into the SCFG produces,~~ for each node of the CCFG, at most one corresponding node in the SCFG <u>is</u> <u>produced</u>.

4. (previously presented) The method of claim 1, wherein the step of scheduling comprises a topological sort for determining the scheduled CCFG.

5. (original) The method of claim 1, wherein an execution of the SCFG comprises translation of the SCFG into a programming language.

6. (original) The method of claim 5, wherein the programming language is C.

7. (original) The method of claim 1, further comprising a step of translation of the SCFG into a programming language.

8. (original) The method of claim 7, further comprising a step of executing the programming language translation of the SCFG.

9. (original) The method of claim 1, wherein an execution of the SCFG comprises interpretation of the SCFG.

10. (currently amended) A data processing system having a memory, comprising the following:

a sub-system configured for inputting a CCFG;

a sub-system configured for scheduling the CCFG to produce a scheduled CCFG;

a sub-system configured for selecting, <u>while a first thread is running,</u> a first node

of the scheduled CCFG;

a sub-system configured for producing a first copy of the first node for an SCFG;

a sub-system configured for determining that the first node is in a second thread, different from the first thread;

a sub-system configured for coupling, to each of a plurality of nodes of the SCFG indicated by the first thread, a node that saves a representation, of a flow of control state of the first thread, in a first state variable, wherein the first state variable is not used to save a representation of a flow of control state of the second thread; and

a sub-system configured for coupling, as a successor node to each node that saves a representation, the same successor node for resuming a flow of control state of the second thread by reading a second state variable and choosing an edge to a node to be executed next in the second thread, wherein the successor node has an outgoing edge to each node that can be executed next in the second thread and the second state variable is not used to save a representation of a flow of control state of the first thread.

~~and~~

~~a sub-system configured for coupling, if a first thread of the first node is suspended, between a second node of the SCFG of a second previously-running thread and the first copy, a first context switch, wherein the context switch saves a second state, of the second previously-running thread, into a state variable dedicated to the second previously-running thread.~~

11. (currently amended) A computer program product comprising a computer usable medium having computer readable code embodied therein, the computer program product including:

computer readable program code devices configured to cause a computer to effect inputting a CCFG;

computer readable program code devices configured to cause a computer to effect scheduling the CCFG to produce a scheduled CCFG;

computer readable program code devices configured to cause a computer to effect selecting, while a first thread is running, a first node of the scheduled CCFG;

Commissioner for Patents
App. No. 09/477,688
September 11, 2004
Page 5 of 10

computer readable program code devices configured to cause a computer to effect producing a first copy of the first node for an SCFG;

computer readable program code devices configured to cause a computer to effect determining that the first node is in a second thread, different from the first thread;

computer readable program code devices configured to cause a computer to effect coupling, to each of a plurality of nodes of the SCFG indicated by the first thread, a node that saves a representation, of a flow of control state of the first thread, in a first state variable, wherein the first state variable is not used to save a representation of a flow of control state of the second thread; and

computer readable program code devices configured to cause a computer to effect coupling, as a successor node to each node that saves a representation, the same successor node for resuming a flow of control state of the second thread by reading a second state variable and choosing an edge to a node to be executed next in the second thread, wherein the successor node has an outgoing edge to each node that can be executed next in the second thread and the second state variable is not used to save a representation of a flow of control state of the first thread.

~~and~~

~~computer readable program code devices configured to cause a computer to effect coupling, if a first thread of the first node is suspended, between a second node of the SCFG of a second previously-running thread and the first copy, a first context switch, wherein the context switch saves a second state, of the second previously-running thread, into a state variable dedicated to the second previously-running thread.~~

12. (currently amended) A computer data signal embodied in a carrier wave and representing sequences of instructions which, when executed by a processor, cause performance of steps of:

inputting a CCFG;

scheduling the CCFG to produce a scheduled CCFG;

selecting, while a first thread is running, a first node of the scheduled CCFG;

producing a first copy of the first node for an SCFG;

determining that the first node is in a second thread, different from the first thread;

coupling, to each of a plurality of nodes of the SCFG indicated by the first thread, a node that saves a representation, of a flow of control state of the first thread, in a first state variable, wherein the first state variable is not used to save a representation of a flow of control state of the second thread; and

coupling, as a successor node to each node that saves a representation, the same successor node for resuming a flow of control state of the second thread by reading a second state variable and choosing an edge to a node to be executed next in the second thread, wherein the successor node has an outgoing edge to each node that can be executed next in the second thread and the second state variable is not used to save a representation of a flow of control state of the first thread.

~~and~~

~~coupling, if a first thread of the first node is suspended, between a second node of the SCFG of a second previously-running thread and the first copy, a first context switch, wherein the context switch saves a second state, of the second previously-running thread, into a state variable dedicated to the second previously-running thread.~~